Journal of Money and Economy
Vol. 18, No. 4, Fall 2023
pp. 497-510
DOI: 10.29252/jme.16.1.1

Original Research Article

Comparing Logistic Regression and LightGBM in Credit Card Fraud Detection: A Statistical Approach Using Prediction Uncertainty

Ramin Mojab[*]

Received: 7 Jan 2025 Approved: 4 Feb 2025

Relying on the Area Under the Curve (AUC) measure, we compare the performance of the Logit regression model and the LightGBM algorithm. Despite these methods being common in the literature, our study emphasizes the role of statistical inference to evaluate and compare the results comprehensively. We use the training set of the Vesta (2023) dataset, provided by Vesta—a global fraud prevention company headquartered in the United States specializing in payment solutions and risk management. Originally released as part of a Kaggle competition focused on credit card fraud detection, this dataset comprises diverse transaction records, representing a rich source for exploring advanced fraud detection methods. Our analysis reveals that while the LightGBM algorithm generally yields higher predictive accuracy, the differences between the calculated AUCs of the two methods are not statistically significant. This underscores the importance of using inferential techniques to validate model performance differences in fraud detection.

Keywords: Fraud Detection, Financial Institution, Credit Card, Logit, LightGBM, Machine Learning
JEL Classification*:* C38, C25, C49

1 Introduction

Card fraud, a subset of identity theft, is a significant type of fraud where individuals' financial card information is stolen and misused (Legal Dictionary, 2023). This includes Card-Present Fraud, such as through theft or counterfeiting, and Card-Not-Present Fraud, which occurs through online payments using card information obtained via data breaches or phishing (European Central Bank, 2021). According to a report by the European

[*] Assistant Professor, Department of Banking, Monetary and Banking Research Institute, Tehran, Iran; rmojab@mbri.ac.ir

Central Bank, the total value of card transaction fraud in 2019 reached 1.03 billion euros, representing 0.032 percent of the total transaction value. Notably, 80 percent of these frauds involved transactions without the physical presence of the card (European Central Bank, 2021).[1]

Card fraud directly harms customers and indirectly harms card-issuing companies. Preventing such fraud involves both the customer and the issuer. On the customer side, education on caring for physical cards and identity information is crucial. On the company side, using up-to-date technology in card issuance (e.g., using chip-based cards instead of magnetic stripes), enacting laws to increase the cost of accessing technologies for producing counterfeit cards, and training or incentivizing vendors to better protect information or identify and prevent fraud are essential measures (Barker et al., 2006).

In recent years, the use of computer algorithms has been proposed as an additional layer for detecting and preventing fraud.[2] The main advantage of these algorithms is their access to very large sample sizes.[3]

In the present study, we examine the role of statistical inference in selecting a fraud detection model. This includes comparing the fraud detection power of logit regression model and the LightGBM algorithm using the Vesta dataset (2023). Additionally, we discuss the feature of financial transaction data, which is characterized by a relatively high volume of observations in the sample. In this context, we also explore the relationship between cross-validation and statistical inference.

The data for the present study is the training set of the Vesta dataset (2023)[4], which was published as part of a competition on Kaggle to find the best fraud prediction model. The LightGBM algorithm plays a prominent role

[1] Shaparak, the backbone of Iran's electronic card payment system, reported a total transaction value of 7000 trillion rials in 2021 (Shaparak, 2022). By applying the same fraud ratio, we estimate an approximate figure of 2.24 trillion rials in fraud. It is important to note that this number is an approximation, considering the type and nature of the transaction process.

[2] For example, Visa has announced that there are eight layers of protection, which include: Visa chip technology, point-to-point encryption, the three-digit code on the back of the card, transaction alerts, device ID (containing information about the device used for the purchase), predictive fraud analysis (where a risk factor is calculated), Verified by Visa (which allows the cardholder to verify the transaction), and tokenization (which allows sensitive account information to be stored as a token). (Visa, 2023)

[3] For example, the number of Shaparak transactions was 33 billion in 2021 (Shaparak, 2022). Additionally, the Vesta dataset (2023) contains 1.1 million observations, covering data from 2017 and 2018, and appears to relate to a six-month period.

[4] Vesta is a fintech pioneer in fraud protection and guaranteed payment technologies, helping online merchants optimize revenue by eliminating the fear of fraud.

[DOI: 10.61186/jme.18.4.497]

in the best fraud prediction models. However, this does not mean that the comparison of fraud detection power between this algorithm and the Logit regression method does not require further investigation. The method of identifying the best model in this competition lacks the desirable characteristics of a statistical study, as there are only two test samples (20% in the public leaderboard and 80% in the private leaderboard), and the significance of the differences between the scores has not been considered. Clearly, in such an approach, sampling error is not managed.

Classifying existing approaches in detecting financial transaction fraud is not simple (for a classification, see Qi (2020)). Some approaches are based solely on transaction information, which is the main focus of this article, where supervised classification algorithms play a central role. In other approaches, historical data on customer behavior is used, and the role of unsupervised algorithms becomes more prominent. Additionally, given the very high volume of requests, discussions about real-time processing or parallel processing arise. On the other hand, operational goals make the approach to fraud detection more systematic and scalable. In any case, as stated, the focus of the present study is on the first approach.

Empirical studies in the field of transaction-based fraud detection differ in terms of the type of data used and feature engineering, the type of algorithm or modeling, and the type of evaluation in the model selection process. In these studies, the issue of fraud detection in financial transactions is usually formulated as a binary classification model, and common machine learning algorithms such as support vector machines, random forests, decision trees, K-nearest neighbors (KNN), naive Bayes classifiers, XGBoost, logistic regression, and/or neural networks are used (for an introduction, see Chaudhary et al., 2012). Given the growth of e-commerce and online payment methods, and possibly as a result of the increase in fraudulent transactions, the number of articles in this field is relatively high, and therefore, only a portion of these recent studies are mentioned here.

Some studies focus on feature engineering, such as Bahnsen et al. (2016) and Zhang et al., (2021). Studies that are closer in subject to the current study are those that examine and compare the predictive power of two or more common learning algorithms as previously presented. Generally, these studies do not lead to a definitive conclusion, and it seems that the issue of selecting the best model depends on the characteristics of the data or the evaluation method. Among these studies are Hossain et al., (2022) [KNN], Dai (2022) [Random Forest], Awoyemi et al., (2017) [KNN], Varmedja et al., (2019), and Shen et al., (2007) [Neural Network and Logistic Regression]. Although there

[DOI: 10.61186/jme.18.4.497]

may be considerations regarding the best model, the best reported model in these studies is presented in brackets (an empty bracket means no specific result was provided). The innovation of the present article, compared to these studies, is the emphasis on the importance of testing the significance of the differences in comparison criteria.

The structure of this article is as follows: Following this introduction, Section 2 delves into the research methodology, emphasizing the approach to comparing model performance. Section 3 introduces the data. Section 4 reports the empirical results, while Section 5 is devoted to discussing the findings.

2 Methodology

A part of the methodology of this article involves estimating Logit regression model and using the LightGBM algorithm. To avoid lengthy discussions, for the first topic, see Greene (2000) and Greene and Hensher (2010), and for the second topic, see Shi et al. (2018) and Chen and Guestrin (2016). Given the importance of model performance evaluation for this article, this section focuses on explaining the AUC criterion. This criterion represents the area under the ROC curve, where the true positive (TP) rate is plotted as a function of the false positive (FP) rate. For details on this, see Fawcett (2006a). The following discussion will focus more on the shortcomings of this criterion and how to address them.

The AUC criterion does not account for the different costs of classification errors. Conceptually, false positives (FP) should have a higher cost than false negatives (FN). If we introduce the following cost matrix:

Table 1
Cost Matrix

		Prediction	
		N	**P**
Actual	N	$C_{TN} = 0.02x$	$C_{FP} = -c$
	P	$C_{FN} = -x$	$C_{TP} = c$

Source: Research Findings

we expect $C_{FP} > C_{TN}$ and $C_{FN} > C_{TP}$. If the first condition is violated and the second condition holds, it is optimal to predict all observations as positive. Conversely, if the first condition holds but the second condition does not, it is optimal to label all observations as negative (Elkan, 2001).

[DOI: 10.61186/jme.18.4.497]

Regarding card fraud, the following statements can be used as the basis for calculating costs: Accepting a fraudulent transaction (FN) costs the value of the transaction (x). Rejecting a valid transaction (FP) annoys the customer and therefore has a fixed cost (c). Rejecting a fraudulent transaction (TP) has a benefit, as it prevents further fraud. This can be considered as the negative fixed cost of a valid transaction (c). Finally, accepting a valid transaction (TN) has a benefit (e.g., 2% of the transaction value), as a service has been provided.

It is possible to calculate the expected cost of the model with this information, but the advantages of the ROC curve are lost. Fawcett (2006b) attempts to address this deficiency by introducing the ROCIV curve and the area under it as a criterion similar to AUC. By setting the origin as 'accepting the transaction,' the benefit of a negative observation (TP_b) is equal to $0.02x + c$, and the cost of a positive observation (FN_c) is equal to $x + c$ (the first column minus the second column, and C_{FN} is converted to cost by negating it). In this case, in plotting the ROC and calculating the AUC, at each threshold level, we move by the amount of TP_b and the cost of FN_c, and normalization is based on the total costs and benefits.

Another shortcoming of AUC is that it includes scenarios for threshold levels that are not practically acceptable. For example, the area where the FP rate is greater than 0.9 means that more than 90% of negative observations are incorrectly classified as positive. Partial AUC is calculated by limiting the ROC curve to operational FP rates. If the area under the curve in this region is divided by the length of the FP rates, this criterion also has a probabilistic interpretation similar to AUC. The generalized form of this correction can be calculated with weighted AUC (Lee and Fine, 2010).

The AUC criterion is not sensitive to the absolute size of the calculated probability for positive and negative observations; what matters is their relative position. For example, as long as all negative group probabilities are above 0.9, but the classification is still correct (the negative group probability for negative observations is greater than the negative group probability for positive observations), this criterion equals one (i.e., maximum). Note that in this example, there is a positive observation with a probability of being positive less than 0.1. In other words, this criterion does not provide information about the optimal or desirable threshold level. This is because this criterion corresponds to the U Mann-Whitney test statistic.

The one-sided version of this non-parametric U Mann-Whitney (-Wilcoxon) test can be used to examine whether one population has a positive or negative shift compared to another population. This statistic is constructed based on combining two samples, sorting them, and examining the rank of

[DOI: 10.61186/jme.18.4.497]

observations from both samples. If the sample size is large enough, the normal approximation can be used to reject the null hypothesis. An important point in this test is that in the sorting process, two or more observations may be equal. This, on the one hand, causes the standard deviation in the normal approximation to be different, and on the other hand, usually a factor of 1/2 is considered in calculating the test statistic. In generalizing this statistic to calculate AUC, the factor of 1/2 is the same situation introduced by Fawcett (2006b, Figure 2) as 'expected.' Choosing a factor of zero is introduced as the 'pessimistic' case, and choosing a factor of one is introduced as the 'optimistic' case.

3 Data

In order to examine the classification power of the Logit regression model and the LightGBM machine learning algorithm in the field of fraud detection, the training sample data from the Vesta dataset (2023) is used. As explained in the introduction, this data was published as part of a competition to find the best fraud prediction model. The study is limited to the training sample because the binary target variable indicating whether the information is fraudulent or not is reported as $isFraud$, and the test sample lacks this column. This sample contains 590,540 observations.

The competition organizer has stated that the labels in the $isFraud$ column are based on the presence of a corrective transaction on the card (Vesta, 2023). If this label is applied to an account, this transaction and other previous transactions with the same email or address will also receive this label ($isFraud = 1$). If no such report exists after 120 days, the transactions are labeled as legal ($isFraud = 0$). In the real world, some fraudulent transactions may not be reported (e.g., due to the cardholder's unawareness or reporting after the legal reporting period). These observations, if they exist, are very unusual and will constitute a very small share of the observations (Vesta, 2023).[1]

The raw data of the training sample includes two tables named identity and transaction. These two tables can be merged using the unique identifier $TransactionID$. The transaction table has 329 columns (excluding $isFraud$ and $TransactionID$). The columns in this table include both quantitative and

[1] In this article, identification processes other than model prediction are not used. For example, based on these explanations, a blacklist of emails or addresses from fraudulent observations is not prepared and is not used in the prediction process. Note that the goal of this article is 'comparison.'

[DOI: 10.61186/jme.18.4.497]

categorical features. These features are explained below (the type of data, i.e., quantitative or categorical, is written in parentheses next to them):

- $TransactionDT$: (Quantitative) Difference in transaction time from a reference time,
- $TransactionAMT$: (Quantitative) Payment amount (USD),
- $ProductCD$: (Categorical) Product code,
- $card1 - card6$: (Categorical) Card information such as card type, card group, issuing bank, country, etc.,
- $addr1, addr2$: (Categorical) Customer address,
- $dist1, dist2$: (Quantitative) Distance between billing address, shipping address, zip code, IP address, phone area, etc.,
- $P\_emaildomain$: (Categorical) Customer email domain,
- $R\_emaildomain$: (Categorical) Recipient email domain (may be empty as some transactions do not have a recipient),
- $C1 - C14$: (Quantitative) Counters, e.g., how many addresses are linked to the card (the actual meaning of these is not clear),
- $D1 - D15$: (Quantitative) Time intervals, e.g., the interval between this transaction and the previous transaction, etc.,
- $M1 - M9$: (Categorical) Matches, e.g., match between names on the card and address, etc.,
- $V1 - V339$: (Quantitative) Features engineered by Vesta, including rankings, counts, etc.

The identity table includes 40 columns (excluding $TransactionID$):

- $DeviceType$: (Categorical) Type of device,
- $DeviceInfo$: (Categorical) Type of device information,
- $id01 - id11$: (Quantitative) Identification-network connection information,
- $id12 - id38$: (Categorical) Information (IP, ISP, Proxy, etc.) and digital signature related to the transaction.

In this research, categorical variables have been converted to dummy variables. Generally, a variable with x categories can be converted to $x - 1$ dummy variables. However, when x is relatively large, it is better to emphasize a few specific categories and use the 'other' option for the rest, or initially group the various categories and then convert them to dummy variables. The final table, which contains only quantitative variables, includes 474 features. However, given that the identity table has fewer observations (144,233 observations), naturally combining the two tables results in a relatively large amount of missing data. Overall, 46% of the numbers in the combined table are NA.

4 Empirical Results

Although different models can be estimated due to the presence of NA and the results of possible models can be combined based on the presence or absence of information in the test sample, since the goal of this article is comparative, two tables without missing observations are designed based on two strategies for removing NA. In the first strategy, the volume of observations is maximized provided that the number of observations removed is as large as possible. This choice results in 590,539 observations and 37 features. In the second strategy, the volume of observations is maximized provided that the number of features is prioritized. This choice results in 44,540 observations and 215 features.

The estimation and evaluation process is not particularly complex and follows a common approach in Monte Carlo cross-validation:

1) A number $N > 0$ is chosen as the number of out-of-sample tests, and a value $t \in (0,1)$ is chosen as the percentage of observations in the training sample.
2) The training sample with a volume of $[tN]$ is obtained randomly and stratified. The remaining observations are called the test sample.
3) The Logit regression model is selected with a top-down approach with a significance threshold level of p_0. This includes estimating the model, removing statistically insignificant variables, and re-estimating the model. Additionally, the optimal parameters of the LightGBM model are selected by designing an out-of-sample evaluation process similar to the current process in the training sample data. The model selection criterion is AUC. At this stage, the weight of positive observations is chosen to be equal to the ratio of total observations to positive observations.
4) The selected models from step 3 are used to predict the test sample observations. The AUC criterion is calculated and stored by selecting the threshold level coefficient α as the economic significance threshold and the 'expected' and 'pessimistic' cases (according to the methodology section).
5) Steps 2 to 4 are repeated N times, and the average results are stored.

The parameters of this simulation are as follows: $N = 200$, $t = 0.6, 0.8$, $p_0 = 0.05$, and $\alpha = 1e-16, 1e-4$. Additionally, the parameters that determine the decision tree design in the LightGBM algorithm are chosen as follows:

[DOI: 10.61186/jme.18.4.497]

- $boosting = gbdt, dart, goss, rf$
- $max\_depth = -1, 3, 6, 12$
- $num\_leaves = 31, 62, 250$
- $learning\_rate = 0.1, 0.05, 0.03, 0.01$
- $bagging\_fraction = 0.5, 0.7, 1.0$
- $feature\_fraction = 0.5, 0.7, 1.0$

It should be noted that not all options are usable in all boosting methods. The parameters assumed to be constant are as follows:

- $nrounds = 500$
- $early\_stopping\_rounds = 20$
- $objective = binary$
- $nthread = 4$
- $bagging\_freq = 2$

Regarding the dart method, since $early\_stopping\_rounds$ is not usable, the number of iterations ($nsample$) is 50. It is important to note that the evaluation criterion in the LightGBM algorithm is also assumed to be AUC, but the corrections mentioned in step 5 are not performed here due to the different process and codes. Further explanations about some parameters are provided, although the main source in this regard is Microsoft (2023).

The different values of the boosting parameter are as follows: the $gbdt$ algorithm, the traditional algorithm in this field proposed by Friedman (2001); rf random forest; $dart$ reduces sensitivity to trees added in the first iterations using neural networks; the $Light$ part in the LightGBM framework naming is due to the $goss$ option, where for efficiency, part of the information with large gradients is removed, and similar items are grouped to reduce feature dimensions.

The max_depth parameter specifies the maximum depth of the tree in each iteration. Increasing the depth will result in better fitting for the training observations, but overfitting needs to be managed. The num_leaves parameter indicates the maximum number of leaves for each tree in each iteration. More leaves result in better in-sample fitting, although overfitting needs to be managed. The $learning\_rate$ parameter adjusts the speed of model formation. Higher values for the $nrounds$ parameter will increase the model training time and the risk of overfitting. This risk is somewhat managed by limiting $early\_stopping\_rounds$ so that if the number of iterations without progress exceeds this value, the iterations stop. The $bagging\_fraction$ value determines what portion of the data is used randomly in each iteration. Similarly, $feature\_fraction$ relates to random

[DOI: 10.61186/jme.18.4.497]

sampling of a portion of the features in each iteration. For more details, refer to Mojab et al. (2022).

The results of the estimation process are reported in Tables 2 and 3 (separately for the two constructed samples)

Table 2
Descriptive statistics of the AUC criterion under different assumptions and in the first NA removal scenario

Training ratio		**60%**		**80%**	
Model		Logit	LightGBM	Logit	LightGBM
Approach	Expected	0.84 (0.02)	0.89 (0.05)	0.85 (0.02)	0.88 (0.03)
	Pessimistic	0.80 (0.02)	0.81 (0.05)	0.81 (0.02)	0.82 (0.04)

Notes: The number of observations is 590,539, and the number of features is 37. The training ratio means the percentage of observations used for training. ‘Approach’ refers to how the AUC is calculated. In each cell, the mean and (standard deviation) of the AUC criteria in all repetitions are reported. *Source*: Research Findings

Table 3
Descriptive statistics of the AUC criterion under different assumptions and in the second NA removal scenario

Training ratio		**60%**		**80%**	
Model		Logit	LightGBM	Logit	LightGBM
Approach	Expected	0.84 (0.02)	0.89 (0.05)	0.85 (0.02)	0.88 (0.03)
	Pessimistic	0.80 (0.02)	0.81 (0.05)	0.81 (0.02)	0.82 (0.04)

Notes: The number of observations is 44,540, and the number of features is 215. For more details, see the footnote of Table 2. *Source*: Research Findings

The definition of significance used in interpreting the results is a rule of thumb: Statistic X with mean and standard deviation μ_X and σ_X and statistic Y with mean and standard deviation μ_Y and σ_Y have a significant difference when $|\mu_X - \mu_Y| \geq 2\sigma_X + 2\sigma_Y$. Almost none of the results are significant.[1] With this assumption, the results show that:

- The absolute comparison of the mean values in Tables 1 and 2 shows that a greater number of features is a more important factor in improving the

[1] The reason for choosing this approximation is that the reported statistics in the table only include variations between groups. A more accurate comparison would consider the variance of each calculated AUC criterion. Since this increases the estimation of variance, the overall results do not change.

average AUC criteria than a greater number of observations. However, it should be noted that the sample size in both scenarios is relatively large (44,000 compared to 590,000 observations, 215 features compared to 37 features). These differences are not significant.

- Comparing the 60% and 80% columns in both tables shows that, in general, increasing the test sample size (or having a smaller test sample size) does not create a significant difference. In some cases, the AUC statistics in the higher test sample size (60% columns) are even higher, although the differences are not statistically significant.
- By comparing the rows of the tables, it can be seen that, as expected, the AUC values in the pessimistic approach are lower.
- Finally, by comparing the columns side by side, we find that, on average, the LightGBM method performed better than the Logit regression method, although this difference is not significant.

5 Conclusion

It seems that in the discussion of fraud detection using data and computational algorithms, the role of statistical inference is overshadowed by operational methods. This operational approach stems partly from the perspective of machine learning literature. However, the large sample size available is a more significant and justifiable factor. In such a large volume of information, there is a heavy reliance on the out-of-sample performance of models. In other words, the high volume of observations allows for the implementation of cross-validation methods. In this method, the validity of the model is inferred from the average out-of-sample prediction errors and not, for example (in a parametric model), from the significance of the parameters. Theoretically, and if the study is limited to linear models, results regarding the equivalence of some types of cross-validation and model selection criteria such as AIC are obtained (e.g., among the early studies are Stone (1976) and Shao (1993 and 1997)). For nonlinear and nonparametric models, the discussion is different, and in general, the goal of prediction and the goal of identifying structure or examining causality need to be separated.

In this article, an attempt was made to conduct a typical study in the field of machine learning with an emphasis on statistical inference. From this perspective, the significance of the superior predictive power of one method over another was emphasized, and an absolute result was avoided (something that was less seen in the reviewed studies). The relatively different (and sometimes contradictory) results of various studies may be attributed to the lack of attention to this feature.

[DOI: 10.61186/jme.18.4.497]

It should be noted that from the perspective of origin, regression analysis pertains to a situation where the role of big data is not yet prominent. When students are introduced to analysis of variance and regression in introductory statistics classes, there is little discussion about the high volume of data and the number of explanatory variables (Clarke et al., 2009). In fact, there, the low degrees of freedom is one of the serious problems. This issue, along with random errors, parameter uncertainty, bias, and misspecification, is considered. However, in the real world, and especially in recent decades, a revolution in terms of computation and data storage has occurred. In this case, the problem of the curse of dimensionality and model uncertainty arises, and nonparametric methods play a special role in analyses. These types of models form the other end of the spectrum. There are no parameters, the class of models is very large, and model uncertainty is the main concern (Clarke et al., 2009). The study results, in terms of comparing average performance, align with this view, although in terms of significance, this result is not confirmed.

What was overlooked in this study is the importance of feature engineering in building a classifier model. In the Vesta competition (2023) (introduced in the introduction), in the public leaderboard (20% of test observations), the AUC value for the top three and thousandth models were 0.968137, 0.967722, 0.967637, and 0.952727, respectively. In the private leaderboard (80% of observations and not visible before the end of the competition), the mentioned value for the top three and thousandth models were 0.945884, 0.944210, 0.943769, and 0.927892, respectively. In terms of the types of models used, the review of high-scoring account reports shows that almost all calculations were based on machine learning algorithms, specifically LightGBM, XGBoost, or Catboost, although neural networks were also occasionally reported (e.g., rank 8 in the private leaderboard). Given the similarity of the models, it seems that the importance of feature engineering in the final performance was very significant.

Finally, it is essential to note that the goal of this study was 'comparison' and not 'maximizing' the AUC value. Accordingly, some information, such as how observations were labeled by Vesta, was not used. For example, a blacklist for rejecting emails or addresses related to fraudulent observations in the training sample was not prepared. Or, for another example, in dealing with missing observations, variables or observations (provided that the total number of observations was maximized) were omitted. Or, for another example, categorical variables were converted to dummy variables in an operational approach, while more information could be extracted from them.

[DOI: 10.61186/jme.18.4.497]

References

Awoyemi, J. O., Adetunmbi, A. O., & Oluwadare, S. A. (2017, October). Credit card fraud detection using machine learning techniques: A comparative analysis. In 2017 international conference on computing networking and informatics (ICCNI) (pp. 1-9). IEEE.

Bahnsen, A. C., Aouada, D., Stojanovic, A., & Ottersten, B. (2016). Feature engineering strategies for credit card fraud detection. Expert Systems with Applications, 51, 134-142.

Barker, K. J., D’Amato, J., & Sheridon, P. (2008). Credit card fraud: awareness and prevention. Journal of Financial Crime, 15(4), 398–410. doi:10.1108/13590790810907236

Chaudhary, K., Yadav, J., & Mallick, B. (2012). A review of fraud detection techniques: Credit card. International Journal of Computer Applications, 45(1), 39-44.

Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining.

Clarke, Bertrand, Ernest Fokoue, and Hao Helen Zhang. (2009). Principles and Theory for Data Mining and Machine Learning. Springer New York, NY. https://doi.org/https://doi.org/10.1007/978-0-387-98135-2.

Dai, S. (2022). Research on Detecting Credit Card Fraud Through Machine Learning Methods. In 2022 2nd International Conference on Business Administration and Data Science (BADS 2022) (pp. 1030-1037). Atlantis Press.

Elkan, C. (2001). The foundations of cost-sensitive learning. In International joint conference on artificial intelligence (Vol. 17, No. 1, pp. 973-978). Lawrence Erlbaum Associates Ltd.

European Central Bank (2021). Seventh report on card fraud. [online]. Available at: https://www.ecb.europa.eu/pub/cardfraud/html/ecb.cardfraudreport202110~cac4c418e8.en.html [Accessed 6 Jan. 2023].

Fawcett, T. (2006a). An introduction to roc analysis. Pattern Recognition Letters, 27(8), 861–74.

Fawcett, T. (2006b). ROC graphs with instance-varying costs. Pattern Recognition Letters, 27(8), 882-891.

Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. Annals of statistics, 1189-1232.

Greene, W. H. (2000). Econometric analysis [5th edition]. International edition, New Jersey: Prentice Hall .

Greene, W. H., & Hensher, D. A. (2010). Modeling ordered choices: A primer. Cambridge University Press.

Hossain, M. N., Hassan, M. M., & Monir, R. J. (2022). Analyzing the Classification Accuracy of Deep Learning and Machine Learning for Credit Card Fraud Detection. Asian Journal for Convergence in Technology (AJCT) ISSN-2350-1146, 8(3), 31-36.

[DOI: 10.61186/jme.18.4.497]

Legal Dictionary (2023). Fraud - Definition, Meaning, Types, Examples of fraudulent activity. [online] Available at: https://legaldictionary.net/fraud/ [Accessed 6 Jan. 2023]

Li, J., & Fine, J. P. (2010). Weighted area under the receiver operating characteristic curve and its application to gene selection. Journal of the Royal Statistical Society: Series C (Applied Statistics), 59(4), 673-692.

Microsoft Corporation (2023). LightGBM. Retrived from: https://lightgbm.readthedocs.io/en/latest/ (at 1 Jan 2023).

Mojab, R., Heidari, H., & Ebrahimi, S. (2022). Design and Determination of Customer Ranking Model for the Export Development Bank. Tehran: Monetary and Banking Research Institute.

Qi, R. (2020). Real-world Credit Card Fraud Detection with Rich Features and Advanced Classification Methods [MSc Dissertation]. School of Computer Science and Informatics. Cardiff University.

Shao, J. (1993). Linear model selection by cross validation. Journal of the American Statistical Association, 88(422), 486–494. https://doi.org/10.1080/01621459.1993.10476408

Shao, J. (1997). An asymptotic theory for linear model selection. *Statistica Sinica*, 7(2), 221–242.

Shaparak (2022). Shaparak Economic Report. Tehran: Shaparak Electronic Payment Network Company. Online access: https://shaparak.ir/, Access date: 01/01/2023.

Shen, A., Tong, R., & Deng, Y. (2007, June). Application of classification models on credit card fraud detection. In 2007 International conference on service systems and service management (pp. 1-4). IEEE.

Shi, Y., Li, J., & Li, Z. (2018). Gradient boosting with piece-wise linear regression trees. arXiv preprint arXiv:1802.05640.

Stone, M. (1976). An asymptotic equivalence of choice of model by cross validation and Akaike’s criterion. Journal of the Royal Statistical Society: Series B (Methodological), 38(2), 276–278. https://doi.org/10.1111/j.2517-6161.1976.tb00932.x

Varmedja, D., Karanovic, M., Sladojevic, S., Arsenovic, M., & Anderla, A. (2019, March). Credit card fraud detection-machine learning methods. In 2019 18th International Symposium INFOTEH-JAHORINA (INFOTEH) (pp. 1-5). IEEE.

Vesta (2023). https://www.kaggle.com/c/ieee-fraud-detection/discussion/101203#589276 (Access date: 1/1/2023).

VISA (2023) Payment Security in Multiple Layers (online). Accessed at: 1/9/2023 (https://usa.visa.com/content/dam/VCOM/Media%20Kits/PDF/PaymentSecurity_Infographic.pdf)

Zhang, X., Han, Y., Xu, W., & Wang, Q. (2021). HOBA: A novel feature engineering methodology for credit card fraud detection with a deep learning architecture. Information Sciences, 557, 302-316.

[DOI: 10.61186/jme.18.4.497]